BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

RLY

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/702,405	10/31/2000	David Hoyle	TI-30561	1217

23494 7590 05/28/2004

TEXAS INSTRUMENTS INCORPORATED
P O BOX 655474, M/S 3999
DALLAS, TX 75265

EXAMINER	
HUISMAN, DAVID J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 05/28/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/702,405	HOYLE ET AL
	Examiner	Art Unit
	David J. Huisman	2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 May 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,2,7,9-12 and 14-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2,7,9-12 and 14-19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *08 March 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-2, 7, 9-12, and 14-19 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: Amendment as received on 5/3/2004.

Specification

3. The disclosure is objected to because of the following informalities: Please remove the attorney docket numbers in the first paragraph on page 1. Also, insert a comma after ".D1" in the second full paragraph on page 14.

Appropriate correction is required.

Claim Objections

4. Claim 2 is objected to because of the following informalities: For increased clarity, please insert --source-- after "second". Appropriate correction is required.

5. Claim 12 is objected to because of the following informalities: In line 2, replace "instruction" with --instructions--. Also, insert --source-- after "second" in line 4. Appropriate correction is required.

6. Claim 17 is objected to because of the following informalities: In line 2, replace "instruction" with --instructions--. Also, insert --source-- after "second" in line 4. Appropriate correction is required.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

8. Claims 7, 12, and 14-17 are rejected under 35 U.S.C. 102(a) as being anticipated by Intel, IA-64 Application Developer's Architecture Guide, May 1999 (as applied in the previous Office Action and herein referred to as Intel).

9. Referring to claim 7, Intel has taught a digital system, comprising a microprocessor, wherein the microprocessor comprises:

a) program fetch circuitry. See page 6-2, section 6.1.1 and note that instructions are fetched by the processor. It is inherent that some circuitry must exist in order to allow for the fetching of instructions.

b) instruction decode circuitry. See page 6-2, section 6.1.1 and note that instructions are decoded by the processor. It is inherent that decode circuitry must exist in order to allow for the decoding of instructions.

c) at least a first functional unit connected to receive control signals from the instruction decode circuitry. See page 6-2, section 6.1.1 and note that instructions are executed by the processor. It is inherent that a functional unit would exist in order to perform such executions.

d) the functional unit comprising byte intermingling circuitry connected to receive a first source operand having a plurality of ordered fields and a second source operand having a plurality of

ordered fields and having outputs connected to provide a destination operand in response to the control signals, wherein the byte intermingling circuitry is operable, responsive to one of a plurality of byte intermingling instructions, to place the contents of a least significant plurality of fields selected from the second source operand in a least significant portion of the destination operand, and to place the contents of a most significant plurality of fields from the first source operand in a most significant portion of the destination operand. See page 7-117 and note the mix1.l instruction. This instruction takes two operands with ordered fields (ordered based on significance) and mixes them. More specifically, the contents of a least significant plurality of fields (fields 5, 3, and 1, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0) of the second operand r2 are written to a least significant portion of the destination operand r1 (fields 5, 3, and 1). The least significant portion of the destination comprises the lower 7 fields (7 through 1) because these 7 fields do not include the most significant field, thereby making them the least significant portion. In addition, the contents of a most significant plurality of fields (fields 7, 5, and 3, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0) of the first operand r3 are written to a most significant portion of the destination operand r1 (fields 6, 4, and 2). The most significant portion of the destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion.

e) Although Intel has suggested the use of a pipeline (see page 11-13, section 11.3.4), Intel has not explicitly taught a plurality of pipeline phases, wherein the fetch circuitry performs a first portion of the plurality of pipeline phases, the decode circuitry performs a second portion of the plurality of pipeline phases and the functional unit performs a third portion of the plurality of

pipeline phases, the third portion being execution phases. However, Hennessy has taught a plurality of pipeline phases, which includes a first fetch phase (IF), a second decode phase (ID), and a third execute phase (EX). See Figure 3.2 on page 132. As can be seen from the Figure's description and from pages 125-126 of Hennessy, pipelining increases the amount of parallelism within a processor. More specifically, instructions can be executed in parallel as opposed to serially. This results in an increase in throughput since a new instruction can be started every cycle as opposed to starting only when the previous instruction ends. As a result, execution speed is increased. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to implement a plurality of pipeline phases in Intel as taught by Hennessy.

f) Finally, it is inherent that the IA-64 microprocessor (which is the type taught by Intel) has an instruction execution pipeline with a plurality of phases. Pipelining is an established technique that allows for parallel execution of instructions as opposed to serial execution, thereby resulting in higher execution speeds.

10. Referring to claim 12, Intel has taught a method of operating a digital system having a microprocessor and a set of byte intermingling instructions, comprising the steps of:

a) fetching a byte intermingling instruction for execution. See page 6-2, section 6.1.1 and note that instructions are fetched by the processor. Note that one of the instructions fetched can be one of the byte-intermingling instructions shown in chapter 7 of Intel (for instance, the first instruction fetched may the "mix1.l" instruction shown on page 7-118).

b) fetching a first source operand and a second operand selected by the byte intermingling instruction, each of the first and second source operands comprising an ordered plurality of

fields. See page 7-116 and 7-117 and note that the first instruction requires two source operands, which have to be fetched. In addition, the operands have ordered fields in that they are ordered based on significance. Clearly, the rightmost field is the least significant field and the fields to the left have increasing significance. See page 7-118 as well for additional confirmation of this significance.

c) writing, into a most significant portion of a destination operand, non-contiguous data from selected ones of the plurality of fields from the first source operand and writing, into a least significant portion of the destination operand, non-contiguous data from selected ones of the plurality fields from the second source operand that are at the same positions as the selected fields of the first source operand, the data being selected in accordance with the byte intermingling instruction. See page 7-117 and note the mix1.l and mix1.r instructions. These instructions take two operands with ordered fields (ordered based on significance) and mix them. More specifically, non-contiguous data from selected fields (fields 7, 5, 3, and 1 for mix1.l and fields 6, 4, 2, and 0 for mix1.r) of the first source operand r2 are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). The most significant portion of the destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion. In addition, non-contiguous data from selected fields (fields 7, 5, 3, and 1 for mix1.l and fields 6,4,2, and 0 for mix1.r) of the second operand r3 are written to a least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). The least significant portion of the destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion. It should be noted that the fields selected from

the two source operands are fields at the same positions (fields 7, 5, 3, and 1 for mix1.l and fields 6, 4, 2, and 0 for mix1.r).

11. Referring to claim 14, Intel has taught a method as described in claim 12. Intel has further taught that the step of writing is performed during a single execution phase of the microprocessor. It should be realized that each instruction will go through a single execution phase, which is the phase in which the instruction is executed. Therefore, when the mix instruction is executed, the writing involved with the mix instruction is performed during the single execution phase for that instruction.

12. Referring to claim 15, Intel has taught a method as described in claim 12. Intel has further taught that the writing step writes most significant bytes of a plurality of fields selected from the second source operand into the least significant portion of the destination operand and writes most significant bytes of a plurality of fields selected from the first source operand into the most significant portion of the destination operand. Again, see page 7-117. Note that most significant bytes from selected fields 7, 5, 3, and 1 of the second operand r3, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to the least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). In addition, most significant bytes from selected fields 7, 5, 3, and 1 of the first operand r2, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). It should be noted that the least significant portion of both the operand and destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion. Similarly, the most significant portion of both the operand and

destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion.

13. Referring to claim 16, Intel has taught a method as described in claim 12. Intel has further taught that the writing step writes least significant bytes of a plurality of fields selected from the second source operand into the least significant portion of the destination operand and writes least significant bytes of a plurality of fields selected from the first source operand into the most significant portion of the destination operand. Again, see page 7-117 and specifically note the mix1.r instruction. Note that least significant bytes from selected fields 6, 4, 2, and 0 of the second operand r3, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to the least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). In addition, least significant bytes from selected fields 6, 4, 2, and 0 of the first operand r2, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). It should be noted that the least significant portion of both the operand and destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion. Similarly, the most significant portion of both the operand and destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion.

14. Referring to claim 17, Intel has taught a method of operating a digital system having a microprocessor and a set of byte intermingling instructions, comprising the steps of:

a) fetching a byte intermingling instruction for execution. See page 6-2, section 6.1.1 and note that instructions are fetched by the processor. Note that one of the instructions fetched can be one of the byte-intermingling instructions shown in chapter 7 of Intel (for instance, the first instruction fetched may the "mix1.l" instruction shown on page 7-118).

b) fetching a first source operand and a second operand selected by the byte intermingling instruction, each of the first and second source operands comprising an ordered plurality of fields. See page 7-116 and 7-117 and note that the first instruction requires two source operands, which have to be fetched. In addition, the operands have ordered fields in that they are ordered based on significance. Clearly, the rightmost field is the least significant field and the fields to the left have increasing significance. See page 7-118 as well for additional confirmation of this significance.

c) writing, into a most significant portion of a destination operand, a most significant plurality of fields selected from the first source operand and writing, into a least significant portion of the destination operand, a least significant plurality of fields selected from the second source operand. See page 7-117 and note the mix1.l instruction. This instruction takes two operands with ordered fields (ordered based on significance) and mixes them. More specifically, a most significant plurality of fields (fields 7, 5, 3, and 1, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0) of the first source operand r2 are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). The most significant portion of both the operand and destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion. In addition, a least significant plurality of fields (fields 5, 3, and 1, where the leftmost

and most significant field is 7 and the rightmost and least significant field is 0) of the second operand r3 are written to a least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). The least significant portion of both the operand and the destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion.

Claim Rejections - 35 USC § 103

15. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

16. Claims 1-2, 9-10, and 18-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Intel, as applied above, in view of Hennessy and Patterson, Computer Architecture - A Quantitative Approach, 2nd Edition, 1996 (as applied in the previous Office Action and herein referred to as Hennessy).

17. Referring to claim 1, Intel has taught a digital system comprising a microprocessor, wherein the microprocessor comprises:

a) program fetch circuitry. See page 6-2, section 6.1.1 and note that instructions are fetched by the processor. It is inherent that some circuitry must exist in order to allow for the fetching of instructions.

b) instruction decode circuitry. See page 6-2, section 6.1.1 and note that instructions are decoded by the processor. It is inherent that decode circuitry must exist in order to allow for the decoding of instructions.

c) at least a first functional unit connected to receive control signals from the instruction decode circuitry. See page 6-2, section 6.1.1 and note that instructions are executed by the processor. It is inherent that a functional unit would exist in order to perform such executions.

d) the functional unit comprising byte intermingling circuitry connected to receive a first source operand having a plurality of ordered fields and a second source operand having a plurality of ordered fields and having outputs connected to provide a destination operand in response to the control signals, wherein the byte intermingling circuitry is operable, responsive to one of a plurality of byte intermingling instructions, to place non-contiguous data from selected fields of the first source operand in a most significant portion of the destination operand, and to place non-contiguous data from selected fields of the second source operand that are at the same positions as the selected fields from the first source operand, in a least significant portion of the destination operand. See page 7-117 and note the mix1.l instruction. This instruction takes two operands with ordered fields (ordered based on significance) and mixes them. More specifically, non-contiguous data from selected fields (fields 7, 5, 3, and 1, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0) of the first operand r2 are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). The most significant portion of the destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion. In addition, non-contiguous data from selected fields (fields 7, 5, 3, and 1, where the

leftmost and most significant field is 7 and the rightmost and least significant field is 0) of the second operand r3 are written to a least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). The least significant portion of the destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion. It should be noted that the fields selected from the two source operands are fields at the same positions (fields 7, 5, 3, and 1).

e) Although Intel has suggested the use of a pipeline (see page 11-13, section 11.3.4), Intel has not explicitly taught a plurality of pipeline phases, wherein the fetch circuitry performs a first portion of the plurality of pipeline phases, the decode circuitry performs a second portion of the plurality of pipeline phases and the functional unit performs a third portion of the plurality of pipeline phases, the third portion being execution phases. However, Hennessy has taught a plurality of pipeline phases, which includes a first fetch phase (IF), a second decode phase (ID), and a third execute phase (EX). See Figure 3.2 on page 132. As can be seen from the Figure's description and from pages 125-126 of Hennessy, pipelining increases the amount of parallelism within a processor. More specifically, instructions can be executed in parallel as opposed to serially. This results in an increase in throughput since a new instruction can be started every cycle as opposed to starting only when the previous instruction ends. As a result, execution speed is increased. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to implement a plurality of pipeline phases in Intel as taught by Hennessy.

18. Referring to claim 2, Intel in view of Hennessy has taught a system as described in claim 1. Intel has further taught that the byte intermingling circuitry is operable to receive the first

source operand and second operand and to provide the destination operand during a single pipeline execution phase. It should be realized that each instruction will go through a single execution phase, which is the phase in which the instruction is executed. Therefore, when the mix instruction is executed, the reading of operands and writing involved with the mix instruction is performed during the single execution phase for that instruction.

19. Referring to claim 9, Intel in view of Hennessy has taught a digital system as described in claim 1. Intel has further taught a register file connected to the first functional unit for providing the first and second source operands and connected to the first functional unit to receive the destination operand. See pages 3-1 and 9-1 and note that the processor contains multiple types of registers. The general purpose registers and floating-point registers would be those used as sources and destinations in the aforementioned operations.

20. Referring to claim 10, Intel in view of Hennessy has taught a digital system as described in claim 1. Intel has further taught each of the set of byte intermingling instructions has a field for identifying a predicate register. See page 9-2 and more specifically the "qp" field in section 9.3.1, which discusses the format of a basic IA-64 instruction.\

21. Referring to claim 18, Intel in view of Hennessy has taught a system as described in claim 1. Intel has further taught that the byte intermingling circuitry is operable to place most significant bytes of a plurality of fields selected from the second operand into the least significant portion of the destination operand and to place most significant bytes of a plurality of fields selected from the first source operand into the most significant portion of the destination operand. Again, see page 7-117. Note that most significant bytes from selected fields 7, 5, 3, and 1 of the second operand r3, where the leftmost and most significant field is 7 and the

rightmost and least significant field is 0, are written to the least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). In addition, most significant bytes from selected fields 7, 5, 3, and 1 of the first operand r2, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). It should be noted that the least significant portion of both the operand and destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion. Similarly, the most significant portion of both the operand and destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion.

22. Referring to claim 19, Intel has taught a system as described in claim 1. Intel has further taught that the byte intermingling circuitry is operable to place least significant bytes of a plurality of fields selected from the second source operand into the least significant portion of the destination operand and to place least significant bytes of a plurality of fields selected from the first source operand into the most significant portion of the destination operand. Again, see page 7-117 and specifically note the mix1.r instruction. Note that least significant bytes from selected fields 6, 4, 2, and 0 of the second operand r3, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to the least significant portion of the destination operand r1 (fields 6, 4, 2, and 0). In addition, least significant bytes from selected fields 6, 4, 2, and 0 of the first operand r2, where the leftmost and most significant field is 7 and the rightmost and least significant field is 0, are written to a most significant portion of the destination operand r1 (fields 7, 5, 3, and 1). It should be noted that the least significant portion

of both the operand and destination comprises the lower 7 fields (0 through 6) because these 7 fields do not include the most significant field, thereby making them the least significant portion. Similarly, the most significant portion of both the operand and destination comprises the upper 7 fields (7 through 1) because these 7 fields do not include the least significant field, thereby making them the most significant portion.

23. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Intel in view of Hennessy, as applied above, in view of Haataja, U.S. Patent No. 6,137,836 (as applied in the previous Office Action).

24. Referring to claim 11, Intel in view of Hennessy has taught a digital system as described in claim 1. Intel in view of Hennessy has not taught that the digital system is a cellular telephone comprising the components set forth in claim 11. However, Haataja has taught a cellular telephone comprising:

a) an integrated keyboard connected to the CPU via a keyboard adapter. See Fig.8, component 72.

b) a display, connected to the CPU via a display adapter. See Fig.8, component 36.

c) radio frequency (RF) circuitry connected to the CPU. See Fig.8, component 56, and column 7, lines 6-11.

d) an aerial connected to the RF circuitry. See Fig.8, component 54.

It should be realized that Intel in view of Hennessy has taught a system that includes operations that increase the functionality of the system. A person of ordinary skill in the art would have recognized that an improved processor (with more functionality) would lead to the overall

improvement of the device in which it is embedded. As shown in Fig.8 of Haataja, and, as is well known in the art, cellular telephones are controlled by some sort of processor. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to have the incorporate the digital system of Intel in view of Hennessy into a cell phone, as taught by Haataja, in order to improve the overall performance of the cell phone.

Response to Arguments

25. Regarding applicants' assertion that the prior art of record does not teach the newly amended claims, the examiner responds by making the above rejections. Within these rejections, it is explained how the prior art of record does in fact read on applicants' claims.

Conclusion

26. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (703) 305-7811. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
May 25, 2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100